

IN THE CLAIMS

1. (Currently Amended) A method for updating software stored in a memory (12) of a mobile device (10), comprising the steps of:

    updating (60) a memory block of the a memory of a mobile device (12) by merging said memory block with differential information from a differential file (21) stored in the memory for updating software stored in said memory (12),

    storing (62) the updated memory block in a backup memory area (32) of the memory (12),

    determining (64) whether the updated block stored in the backup memory area (32) is correct, and

    copying (66) the updated block from the backup memory area (32) to an original location, if the updated block is correct.

2. (Currently Amended) The method of claim 1, wherein if the updated block is correct, further comprising the step of:

    writing (66) a new block status.

3. (Currently Amended) The method of claim 1, wherein the software to be updated is located in a software image area (18) of the memory (12).

4. (Currently Amended) The method of claim 1, wherein the software to be updated is located in a variant package area (16) of the memory-(12).

5. (Currently Amended) The method of claim 1, wherein the differential file (21)-is installed and stored in a user file system area (17)-of the memory-(12).

6. (Currently Amended) The method of claim 1, wherein the differential file (21)-is downloaded by a user.

7. (Currently Amended) The method of claim 1, before the step of updating (60)-a memory block of the memory-(12), further comprising-the step-of:

checking (42)-validity of an update-application stored in the memory-(12), said update-application is used-for facilitating said updating-(60).

8. (Currently Amended) The method of claim 7, wherein the update-application is stored in an update-application area (30)-of the memory (12)-and in a backup area (32)-of the memory-(12).

9. (Currently Amended) The method of claim 7, wherein the update-application is valid, and before the step-of updating (60)-the software, further comprising-the steps of:

checking (52)-if the differential file (21)-contains data for updating the software, and

reading (56)—the data for updating the software from the differential file (21)—if said data is available.

10. (Currently Amended) The method of claim 9, further comprising the steps of:

determining (58)—if there is a further block that needs to be updated by identifying a last updated block from a status, and

writing (68)—new checksums for an updated software if there is no the further block to be updated.

11. (Currently Amended) The method of claim 9, wherein the update-application is valid, and before the step of updating (60)—the software, further comprising the steps of:

checking (52)—if the differential file (21)—contains information for updating the update-application, and

updating (74)—the update-application, verifying (76)—it and writing (80)—a new checksum for the updated update-application, if the differential file (21)—contains information for updating the update-application.

12. (Currently Amended) The method of claim 9, wherein checking (42)—the validity of the update-application is verified by comparing a checksum or a backup checksum generated for an update-application stored in an update-application area (30)—of the memory (12)—or in a backup area (32)—of the memory (12), respectively, with an original checksum stored in the memory (12)—to verify that both checksums are identical.

13. (Currently Amended) The method of claim 12, wherein the original checksum is stored in an update-application checksum area ~~(34)~~—of the memory—~~(12)~~.

14. (Currently Amended) The method of claim 12, wherein the checksum and the original checksum are not identical but the backup checksum and the original checksum are identical, further comprising—the step of:

writing the update-application from the backup area ~~(32)~~—to the update-application area ~~(30)~~.

15. (Currently Amended) A memory ~~(12)~~—of a mobile device ~~(10)~~, comprising:

an update-application area ~~(30)~~—for storing an update-application for updating software of the memory ~~(12)~~;

a backup area ~~(32)~~—for temporarily storing the memory block that is updated; and

an update-application checksum area ~~(34)~~—for storing the checksums.

16. (Currently Amended) The memory ~~(12)~~—of claim 15, wherein the update-application area ~~(30)~~, the backup area ~~(32)~~ and the update-application checksum area ~~(34)~~—are located in an update means area ~~(28)~~—of the memory ~~(12)~~.

17. (Currently Amended) The memory ~~(12)~~—of claim 15, further comprising a differential file ~~(21)~~—for updating the software of the memory ~~(12)~~.

18. (Currently Amended) A method, ~~for updating software stored in a memory (12) of a mobile device (10)~~ comprising the steps of:

    checking ~~(42)~~ validity of an update-application stored in ~~the-a~~ memory of a mobile device (12), and

    updating ~~(60)~~ the software stored in the memory using a block-by-block approach based on differential information from a differential file ~~(21)~~ downloaded to and stored in the memory ~~(12)~~—if the update-application is valid, wherein said update is done by overwriting a block with the differential information at a location in the memory ~~(12)~~ that is different from an original memory location of said memory block in the memory—~~(12)~~, wherein said update-application is ~~used~~ for facilitating said updating—~~(60)~~.

19. (New) The method of claim 18, wherein the differential file is installed and stored in a user file system area of the memory.

20. (New) The memory of claim 15, wherein said checksums comprises a checksum for said update application.

21. (New) A memory of a mobile device, comprising:

    storing means, for storing an update-application for updating software of the memory;

    further storing means, for temporarily storing the memory block that is updated; and

    still further storing means, for storing checksums for said update application.

944-005.015  
Serial No. 10/688,210

22. (New) The memory of claim 21, wherein the storing means is an update-application area for storing, the further storing means is a backup area for said temporarily storing, and the still further storing means is an update-application checksum area.